

AMENDMENTS TO THE CLAIMS

Please amend the claims without prejudice. The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of the Claims

1. (Currently amended) A system for near real-time transfer of a datafile from a first computer to a second computer without requiring extensive protocol customization on the second computer and, comprising:

a first computer having:

a connection to a computer network and operable to communicate over the computer network using a standard protocol;

a server side script, responsive to a down-load request from a second computer, operable to launch an http data streamproducer of the standard protocol and to read and write data over the computer network using the standard protocol;

the http streamproducer operable to read a ~~designed~~ designated source file and simultaneously write data from the source file into a return-data-buffer connected to the server-side script; and

a read-while-write mechanism allowing the http streamproducer to read data from the designated source file while the designated source file is being written by a data producer program; and

a second computer having:

a connection to the computer network and operable to communicate over the computer network using the standard protocol;

a transaction handler class, each instance of which is operable to read and write data over the computer network using the standard protocol and to write blocks of data to a destination simultaneously with receiving data from the computer network; and

a data StreamHandler for interpreting a database stream received from the transaction handler.

2. (Original) The system of claim 1 wherein

the first computer further comprises:

a webserver for transmitting a webpage containing a list of files available for download by other computers;

the second computer further comprises:

a webbrowser for displaying the webpage containing the list of files available for download; and

a trusted applet operable, in response to a user selecting a file from the list, to create a transaction handler instance for receiving the selected file.

3. (Previously presented) The system of claim 1 wherein

the second computer further comprises:

at least one stream handler class having at least one file interaction method for performing a file operation selected from the set of steps comprising creating a file, opening a file and writing to a file; and

wherein the transaction handler instance creates a stream handler instance appropriate for the file selected by the user.

4. (Original) The system of claim 1 wherein the standard protocol is http.

5. (Original) The system of claim 1 wherein the standard protocol is WAP.

6. (Previously presented) The system of claim 4 wherein the server-side script implements an http GET command and the down-load request is an invocation of the http GET command of the server-side script.

7. (Previously presented) The system of claim 4 further comprising an `HttpStreamProducer` class and wherein the `HttpStreamProducer` is an instance of the `HttpStreamProducer` class.

8. (Original) The system of claim 1 wherein

the first computer further comprises:

a webserver for transmitting a webpage containing a list of files for download by other computers;

the second computer further comprises:

a webbrowser for displaying the webpage containing the list of files available for download; and

a trusted applet operable, in response to a user selecting a file from the list, to create a transaction controller instance operable to manage a plurality of file transfer threads, wherein in each file transfer thread, in response to the request from a user to download a file, the transaction controller instance is operable to create a transaction handler instance for receiving data from the first computer.

9. (Previously presented) The system of claim 8 wherein

the second computer further comprises:

a data stream handler class having a method for receiving data from the transaction handler instance and for writing data to a destination.

10. (Original) The system of claim 9 wherein the destination is a data file.

11. (Original) The system of claim 9 wherein the destination is an application program that is a data consumer.

12. (Original) The system of claim 11 wherein the destination is a database.

13. (Canceled)

14. (Canceled)

15. (Previously presented) A method for near-real time download of a file via a computer network, comprising:

- a) on a server producing a list of files available for download;
- b) on a client retrieving the list of files available for download;
- c) selecting a file from the list;
- d) in response to the selection of a file from the list, creating a transaction handler instance, wherein each transaction handler is operable to read and write data over the network;
- e) operating the transaction handler instance to transmit a request over the computer network indicating to the server to transmit the selected file;
- f) receiving the request at the server;
- g) in response to receiving the request at the server:
 - reading blocks of data from the selected file;
 - placing the blocks of data in a return buffer;
 - transmitting the blocks of data from the return buffer to the client concurrently with reading additional blocks of data;
- h) receiving the blocks of data at the transaction handler at the client;
- i) transferring the blocks of data to a data streamhandler application; and
- j) writing the blocks of data to a destination file concurrently with receiving additional blocks of data.

16. (Previously presented) The method of claim 15 wherein said blocks transferring step further comprises:

k) launching an application program, the application program being the sata streamhandler; and

l) upon receipt of blocks of data at the client, transferring the blocks of data to the application program.

17. (Original) The method of claim 15 wherein the step (a) of producing a list of files available for download comprises the step of creating a web page including the list of files.

18. (Original) The method of claim 17 further comprising the step of transmitting from the server to the client the web page including the list of files.

19. (Currently amended) An article of manufacture comprising a program storage medium having computer readable program code means embodied therein, wherein the computer readable program code comprises instructions to cause a computer system, having a server side computer, a client side computer, and a computer network connecting the server side computer to the client side computer to:

produce a list of files available for download from the server side computer;

display the list of files available for download on the client side computer;

allow a user to select one or more of the files available for download;

in response to the selection of a file from the list, create a transaction handler instance, wherein each transaction handler is operable to read and write data over the network;

transmit a request over computer network indicating to the server to transmit the selected file;

receive the request at the server;

in response to receiving the request at the server:

read blocks of data from the selected file;

place blocks of data in a return buffer;

transmit the blocks of data from the return buffer to the client concurrently with reading additional blocks of data;

receive the blocks of data at the client; and

to write the blocks of data to a destination concurrently with receiving additional blocks of data.

20. (Original) The article of manufacture of claim 19, wherein the destination is a computer file.

21. (Original) The article of manufacture of claim 19, further comprising computer readable program code instructions to cause the computer system to: launch an application program on the client side; and wherein the destination is the application program.

22. (Original) The article of manufacture of claim 19, wherein the destination is a database.

23. (Original) An article of manufacture comprising a program storage medium having computer readable program code means embodied therein, wherein the computer readable program code comprises instructions to cause a computer system, having a server side computer, a client side computer, and a computer network connecting the server side computer to the client side computer, to transfer selected files from the server side to the client side, the instructions comprising:

a web page producer;

a web page reader, wherein the web page reader is operable to receive and to display a web page from the web page producer;

a server side script operable to receive a download request and to launch an httpstreamproducer and to receive and transmit data over a standard protocol;

an httpstreamproducer class each instance of which being operable to read a designated source file and simultaneously write data from the source file to a return-data-buffer; and

a read-while-write mechanism providing the computer system instructions to enable the simultaneous reading from and writing to a data source;

wherein the server script is operable to read data blocks from the return-data-buffer and to transmit the data blocks over the computer network;

a transaction controller operable to receive a create instruction and in response to the create instruction, to create a transaction handler;

a transaction handler operable:

to create an httpstreamhandler;

to transmit get commands to a server side script;

to receive blocks of data from the server side script; and

to transfer the data to the httpstreamhandler;

an httpstreamhandler operable:

to receive data from the transactionhandler; and

to write data to a destination.

24. (Original) A system for near real-time transfer of a datafile from a first computer to a second computer, comprising:

a first computer having:

a connection to a computer network and operable to communicate over the computer network using a standard protocol;

a server side script operable to receive download requests from a second computer and, responsive to each download request from the second computer, operable to launch an httpstreamproducer and to read and write data over the computer network using the standard protocol;

each httpstreamproducer operable to read a designated source file and simultaneously write data from the source file into a return-data-buffer connected to the server-side script; and

a read-while-write mechanism allowing the httpstreamproducer to read data from the designated source file while the designated source file is being written by a data producer program;

wherein the server side script is further operable to transmit blocks of data from the plurality of httpstreamproducers over the connection; and
a second computer having:

a connection to the computer network and operable to communicate over the computer network using the standard protocol;

a transaction controller operable to send data to and receive data from the server side script, and further operable to marshall the data to an appropriate transaction handler; and

a transaction handler class, each instance of which is operable to read and write data over the computer network using the standard protocol and to write blocks of data to a destination file simultaneously with receiving data from the computer network.